Using Mobile Sinks in Wireless Sensor Networks to Improve Building Emergency Response

T. T. Truong[1,2], K. N. Brown[2],C. J. Sreenan[1]
[1]Mobile & Internet Systems Laboratory and [2]Cork Constraint Computation Centre,
Department of Computer Science, University College Cork, Ireland.
{tt11, k.brown, cjs}@cs.ucc.ie

***Abstract*-We propose an opportunistic routing scheme for wireless sensor networks operating in volatile environments. In particular, we consider a sensor field for sensing and reporting on buildings during fires, where sensors are progressively being destroyed by the fire. We envisage firefighters equipped with small computers which can act as mobile sink nodes, offering transient shorter routes for relaying data, and offering connectivity to disconnected areas of the network. We examine different ways in which these uncontrolled mobile sinks could enhance performance, and develop protocols for advertising the presence of the mobile sinks, gathering data for forwarding, and prioritising disconnected regions. We evaluate the performance in simulation, and on randomly damaged networks, we show that we can increase the data delivery by up to 50%.**

I. INTRODUCTION

Wireless sensor networks (WSNs) are formed from sensor nodes with limited resources that are deployed to detect physical phenomena. These nodes generate data and operate in a multi-hop fashion to relay data from other nodes. In our case, we consider relaying data to a base station (static data sink) in buildings during a fire as could be used for monitoring the spread of the fire, locating people in the building, and providing real time information to firefighters, etc. This needs robust and rapid communication, yet the sensor field may become unreliable as nodes are consumed by the fire. We envisage firefighters entering the building each with a small powered node attached to them as part of their equipment pack. These nodes can act as mobile sink nodes which are able to relay data to the base station in a single hop, using for example IEEE 802.11. The main question we consider is how to make best use of these mobile sinks in order to improve the efficacy of network delivery. This raises several key research questions. When should sensor nodes relay data via the mobile sink? How does the mobile sink make its presence known to the sensor nodes? How can we use the mobile sink to re-connect disconnected regions of the field? Note that the movement of the mobile sink is not under the control of the WSN. We do not assume that we can direct the firefighter, and so from the point of view of the WSN, the mobility is uncontrolled. In this paper, we develop an opportunistic routing scheme for taking advantage of these uncontrolled mobile sinks in fire systems in buildings: the mobile sinks can collect data locally, or can act as connectors to the disconnected areas. We evaluate the performance using simulation and show that use of the mobile sink can increase the message delivery rate by up to 50%.

The rest of the paper is organized as follows. We survey prior work in Section II. We present our approach in Section III, based on four scenarios of using the mobile sinks. Section IV presents simulation results, and Section V concludes. Due to space restrictions we are unable to present many details of our approach, and can provide only a limited set of results. Interested readers are referred to [1] which includes a complete description of our solution and experimental evaluation.

II. RELATED WORK

Several papers have considered the use of mobile relays to alleviate the problem that nodes close to a base station tend to quickly deplete their energy [2,3,4]. In [4] the authors distinguish between a mobile relay and a mobile sink. The mobile relay will 'pick up' data (when it's close to the sensor nodes) and 'transport' the data by mechanical movement, rather than transmitting it through wireless links. In this case, the data delivery latency is significant. In contrast, a mobile sink (1) distributes load, (2) collects data continuously, and (3) moves slowly and discontinuously [4].

Regarding sink mobility there have been many papers, focusing on three types of movement: random, predictable/deterministic and controlled. Researchers have proposed many solutions for routing and controlling the trajectory of the mobile sinks. For example in [3], the authors define an algorithm that moves the sink to the new site with highest residual energy to balance the energy spent in the network.

In contrast, we revisit the idea of collecting data with a mobile sink but in different context: mobile sinks could be used in fire systems in large buildings, and we propose using *uncontrolled* mobile sinks – although our mobile sinks do provide wireless connectivity, their movement is not directed towards this. Instead, they move freely through the sensor field for some other purpose (fire fighting/rescue), and the network must *opportunistically* use the connectivity they provide.

There are many papers regarding the use of WSNs for building and forest fires, e.g. [5,6], but these do not consider the use of mobile sinks as a solution for improving message delivery in the face of highly unreliable networks.

III. ASSUMPTIONS AND APPROACH

A. Assumptions

Firstly, we have some assumptions about sensor nodes and the base station (BS).

1. There is a static BS and many stationary sensor nodes.

2. Each sensor transmits data back to the BS through a multi-hop path. Each sensor has the same maximum transmission range, and is aware of its "relative" location (within the building).

We also have assumptions about the mobile sink (MS).

3. The MS is aware of its own position, relying on well-known localisation mechanisms such as [7].

4. The MS moves uncontrollably but part-predictably through the field.

5. The MS can detect its speed and direction of travel.

For simplicity, we assume that the original routing tree for routing data from sensor nodes to the BS in the absence of a mobile sink is constructed using literature standard routing protocols, e.g. [8,9].

B. Approach

Our approach is explained in the context of four scenarios in which a mobile sink may be used during a building fire emergency. Table 1 summarises the notation.

K_{BSi}	The hop count to reach the BS from sensor node i
K_{MSi}	The hop count to reach the MS from sensor node i
K_{join}	The value to decide if sensor should join the MS-tree (join if $K_{BSi} > K_{MSi} + K_{join}$)
K	The number of hop counts each beacon should be extended from the MS.
RE_Threshold	The remaining energy threshold to join and leave the MS-tree
SPEED_Threshold	Speed threshold of the MS for issuing a beacon broadcast.
N_K	The average number of nodes within K hops
N_{K-1}	The average number of nodes within K-1 hops
P_{rx}	The power consumption for receiving a message
P_{tx}	The power consumption for transmitting a message
d	The average number of neighbours of each node
T_i	The transmission range equal for each node
S	The 2D region in which the sensors are deployed
N	The total number of sensor nodes

Table 1. Notation

* **Stationary**: Figure 1 shows the case when the MS arrives at a new location in the sensor field, and offers a shorter route to the BS for nodes in its immediate area. MS broadcasts a beacon message, and this message floods though the network for up to K hops. Each sensor that can hear the beacon decides whether it is better to route via the MS (and continue flooding the beacon), or to continue with its old route to the BS. We assume nodes will join the temporary MS-tree, which is rooted at the MS, as soon as they find the MS-tree offers a shorter path. Note that re-routing for a small improvement may be more expensive than keeping the original route due to the overhead of building up the MS-tree, and collapsing it when the MS is out of range.

To estimate the cost of building the MS-tree, we look at the best case when each node within (K-1) hops receives and broadcasts the beacon once and the nodes at exactly K hops only receive the beacon messages without broadcasting them. In this case, the cost of building the MS-tree in K hops is minimal. The energy cost to build the MS-tree is subjected to Eq.1.

Assuming the nodes are uniformly deployed and the area of K-hop neighbourhood of a node is covered by the area of the circle centred at the node with radius KT_i, the average number of nodes within K hop can be estimated simply as Eq.3.

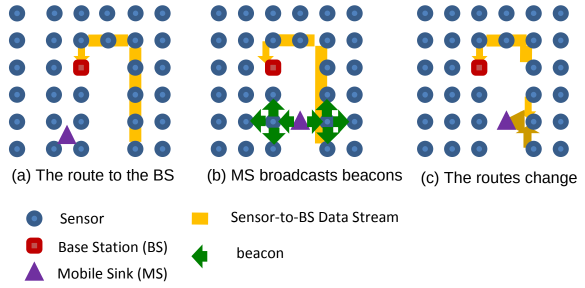


Fig. 1. MS arrives at a new location in the sensor field.

However, when the goal is to consider energy efficiency, the energy cost and/or the residual energy metrics are used. Sensor nodes will decide to set up or join the MS-tree if the power consumption it offers is less than the original route. For load balancing, the metrics can be data rates, transmission rates, etc. Sensor nodes in this case choose the light way to transmit data. In this paper, we combine two metrics: the residual energy and hop count value.

To join the MS-tree, a sensor node that receives the beacon (within K hops from the MS) will compare its local hop K_{BSi} (the hop count to reach the BS) and its new hop K_{MSi} (the hop count to reach the MS). In case that $K_{BSi} > K_{MSi} + K_{join}$, the node will join the MS-tree and forward the beacon to its neighbours if $K_{MSi} < K$. Otherwise, it ignores the beacon message.

$$E \geq N_K.P_{rx} + N_{K-1}.P_{tx} \qquad (1)$$

$$N_K \approx \pi.(KT_i)^2.(N-1)/S \qquad (2)$$

$$N_K = K^2.d \qquad (3)$$

$$d = \pi.T_i^2.(N-1)/S \qquad (4)$$

Sensor nodes can decide to leave the MS-tree if their power is running low (due to relaying/forwarding data to the MS). (i.e. if its residual energy is lower than RE_Threshold). To do that, the node broadcasts a beacon with hop count INFINITY. The children can then find their new parents or leave the MS-tree. When the MS is about to move, it again broadcasts a beacon. How to collapse or revise the tree is discussed below.

* **Movement:** The MS is moving, while acting as a relay (routing data back to the BS). In this scenario, the BS collects data from sensors and MSs. The MS has knowledge of its own velocity and has a strategy for sending the *beacon*. When the MS is moving, the hop count for each sensor node may change frequently. To deal with this, the node will

follow the collapsing policy to decide whether to connect to a new parent or leave the MS-tree.

The collapsing policy: When a node in the MS-tree receives no beacon from the MS, or a beacon with hop count of INFINITY, it firstly sends a warning message to its descendants, saying that the MS-link might be broken. The sensor nodes will wait for a backoff time (K_{MSi}).L/R_i (L is length of beacon message, R_i is data rate). During this time, they will stop forwarding data and store data internally if necessary. If a sensor node receives any beacon message within the backoff time, it will join the new parent if $K_{BSi} > K_{MSi} + K_{join}$, then it forwards its new K_{MSi} to the neighbours. In case that $K_{BSi} \leq K_{MSi} + K_{join}$, the node will revert to the original path to the BS. If it does not receive a beacon in that period, it also wipes out its MS-tree information, and reverts to the standard tree.

The MS has known of its own velocity and direction, and has a strategy for sending the *beacon*. If its speed exceeds *SPEED_Threshold*, the MS stops broadcasting the *beacon*, or broadcasts a beacon with INFINITY hopcount which indicates that it will not be able to receive any data. This strategy is to prevent data losses, and to save energy consumption for sensor nodes in neighbourhood.

* **Reservation:** In a building, the MS can predict the direction of movement in some cases such as when the MS moves along corridors. Then, it broadcasts the RESERVATION message at a time t1. This message will include the predicted location and the time t2 at which the MS is expected to arrive. Ideally this would be achieved using a directional antenna on the MS, as shown in Figure 3, but otherwise we can apply geographic routing, as in GPSR [12].

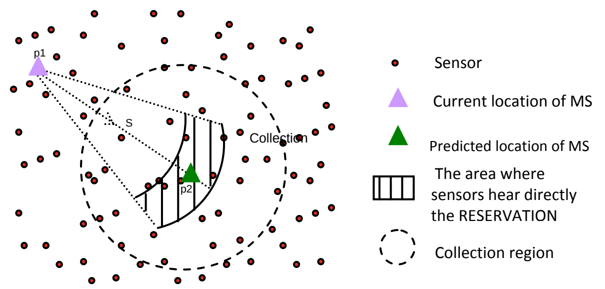


Fig. 2. MS broadcasts where it is going to be.

When a sensor node receives a RESERVATION message, if it is in the collection region, it will join the MS-tree and prepares data to for the MS. In this scenario, we have to deal with the case that the MS is delayed, or never reaches its planned destination. Sensor nodes must decide how to prepare data for the MS arrival, whether it sends data immediately to the MS, or continue to send data on the original path to the BS until some time t1' (t1≤t1'≤t2). If t1' is close to t2, when the MS arrives, the data might not be ready. However, if t1'=t1, sensor nodes will send data to the MS immediately using the MS-tree. When the MS' arrival is delayed, the sensor nodes in collection region will wait for a while before they collapse the MS-tree, and send all data back to the BS via the original path. During this wait time, if they hear from the MS, the collection will be performed normally until the MS moves out of the area.

* **Connection:** Since we assume the WSN is operating in a volatile environment, where some nodes are being destroyed, it is likely that some clusters of nodes will become disconnected from the rest of the network. The mobile sink offers a temporary connection. In each of the three modes described above, we need to adjust the behaviour of the MS and other nodes to recognize and give priority to disconnected regions. In addition, the disconnected nodes need to organize their data collection to be able to take advantage of this transient link if it appears. We assume that once a cluster recognises it is disconnected, its implements a new policy for storing data, discarding less important data, and stopping transmission of data once energy becomes depleted.

IV. Evaluation

In order to evaluate our approach we designed an opportunistic routing protocol and implemented it on (a) a small laboratory-based WSN and (b) within the popular ns-2 network simulator. Details are given in [1], and here we present just the key simulation results. In the simulation, 150 nodes are distributed in a grid area of 10-meter cell length; 149 sensor nodes are placed at crossing points of the grid; the location of the base station (BS – node 0) is fixed at the bottom-left corner of the network map. Hence, the maximum number of hops in the original network using the standard protocol is 25. Table 2 shows the simulation parameters. The transmission range is 10 meters; hence each node can talk to 4 neighbours (left, right, up, down). Each sensor node will sense data and transmit to the BS every Pi seconds.

Parameters	Default Value
Packet size	50 bytes
Transmission range	10 meters
Data Period, Pi	Varied in {5, 10, 20, 30} seconds

(a) ns-2 parameters

RE_Threshold	100 J
K_{join}	2
K	Varied from 5 to 10
Pause time of the MS	Varied in {10, 15, 20, 25, 30}
SPEED_Threshold	4 meter/second

(b) Protocol parameters

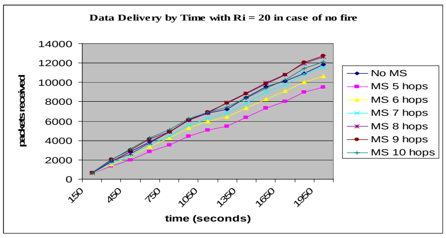
Table 2. Configuration of simulation

For modeling the hazardous environment we are not concerned with faithfully modeling a spreading fire (for example); instead, we simply wish to create isolated regions, and to puncture holes in the network. Thus we simply create rectangles of nodes and disable the nodes on the perimeter. We pick two random coordinates for the bottom left and top right corners of the rectangle, and then disable all the nodes on the boundary of that rectangle following a simple policy. To disable a node, we generate a random time point where the node will die. And then for each step, we check if any nodes reach their time points, and those nodes will be turned off indicating that they are dead. We introduce a mobile sink walking into the simulation area with the Random Way Point Model supported in ns-2. The reservation is made randomly as presented in scenario 3 by the mobile sink (MS).

We are interested in the number of packets successfully received by the BS, and their latency. In the following we will show the impact of the MS though number of packets the BS can receive successfully by time when we use and do not use the MS. With this, we can see the delivery latency, how fast the BS can received data with the aid of the MS. In this simulation, we vary the K hop value from 5 to 10 to see the different benefits from the MS with different collection area sizes. In the network with maximum number of hops is 25, the smallest K with 5 hops is reasonable.

We first measure and compare the data delivery time in normal behaviour when no fire occurs. Figure 3(a) shows the number of packets received by the BS over time when no MS is used and when we introduce a MS with K hops collecting data. Here, we vary the K parameter from 5 to 10 hops. In this experiment, we see that the MS doesn't help at all if K is 5, or 6, or 7. When we increase K with 8, or 9 hops, the mobile sink offers a faster delivery data to the BS. However, with K increased to 10 hops, the data delivery is approximately the same as the case when not using a MS. The introduction of the MS seems to give mixed results. We believe this is explained by the MS occasionally moving too close to the BS, and so creating an overhead in messages without offering any faster route. If the k value is too high, the MS tree is too large, and some data is forwarded to the MS only to arrive after the MS has already departed, and the data has to be re-routed back to the base station, thus increasing latency.

We now introduce the spreading hazard into the network and we vary the K value from 5 to 10. Figure 3(b) shows that the MS gives a significant impact in data delivery when fire occurs with the biggest benefit obtained when K is 8 or 9. With our fire spread model, some nodes will die gradually. The points when the data delivery without the MS changes dramatically, approximately t=840s, is when the network starts to become disconnected. We see the benefit of the mobile sink being able to offer connectivity. At that point, the delivery rate for the No MS situation decreases, and is soon overtaken by the different MS cases. Note that the gradient of the data delivery lines in the Figure 3(b) change at some points, for instance, MS with 5 hops at 1740s, or MS with 10 hops at 1300s, etc. This is due to the difference in the amount of data collected when the MS is moving compared to when it is paused. Overall, we can see that the gives a significant benefit in data delivery. At time = 1950s, the MS increases by approximately 50% the amount of data received by the BS.



(a) Normal behaviour

(b) Emergency behaviour

Figure 3. Data Delivery by Time with Pi=20

V. Conclusion

We have presented a scheme for opportunistically using an uncontrolled mobile sink to achieve reliable and robust data delivery in wireless sensor networks during building emergencies. Our experiments show that with the reservation technique, use of a mobile sink yields increased message delivery rate by up to 50%. Current work includes completing our sensor node software implementation, while future work will include mathematical analysis and extensions for dealing with multiple mobile sinks.

Acknowledgment

The NEMBES project is funded by the Irish Higher Education Authority PRTLI-IV programme.

References

[1] T.T. Truong, K.N. Brown, C.J. Sreenan. "ORMS: Opportunistic Routing with Mobile Sinks for Wireless Sensor Networks", *Tech. Report UCC-CS-2010-03-19, Dept. of Computer Science, University College Cork*, Mar. 2010.

[2] C-h. Ou, K-f. Ssu, "Routing with mobile relays in opportunistic sensor networks", *18th IEEE Int'l Symposium on Personal, Indoor and Mobile Radio Communications,* PIMRC 2007.

[3] S. Basagni, A. Carosi, E. Melachrinoudis, C. Petrioli, Z. M. Wang, "Controlled sink mobility for prolonging wireless sensor networks lifetime", *Journal on Wireless Network,* 2007.

[4] J. Luo, J. Panchard, M. Piorkowski, M. Grossglauser, J-P. Hubaux, "MobiRoute: Routing towards a Mobile Sink for Improving Lifetime in Sensor Networks", *IEEE/ACM DCOSS*, 2006.

[5] P. Ji, C. Wu, Y. Zhang, H. Chu, "A wireless sensor network routing protocol for fire estimation in building," *2nd IEEE Int'l Conf. on Computer Science and Information Technology*, pp.154-158, 2009.

[6] B. Kosucu, K. Irgan, G. Kucuk, S. Baydere, "FireSenseTB: a wireless sensor networks testbed for forest fire detection, *Proc. of 2009 Int'l Conf. on Wireless Communications and Mobile Computing,* 2009.

[7] A. Baggio, K. Langendoen, "Monte Carlo localization for mobile wireless sensor networks", Elsevier Science Pub., pp. 718-733, 2008.

[8] C. Liang, X Huang, J. Deng, "A Fault Tolerant and Energy Efficient Routing Protocol for Urban Sensor Networks**",** *Proc. of 2nd Int'l Conf. on Scalable Information Systems,* 2007.

[9] A. Boukerche, R. Werner N. Pazzi, R.B. Araujo, "Fault-tolerant wireless sensor network routing protocols for the supervision of context-aware physical environments", Elsevier Science Pub., 2006.